

Appln No. 09/618,965

Amdt date May 28, 2004

Reply to Office action of April 2, 2004

**Amendments to the Claims:**

This listing of claims will replace all prior versions, and listings, of claims in the application:

**Listing of Claims:**

---

B1 1. (Currently Amended) A method for interactive debugging in a multi-channel, multi-service system, the method comprising:  
dynamically allocating a plurality of services to a plurality of processors in the multi-channel, multi-service system for execution;

selecting a target construct from the dynamically allocated executing plurality of services for debugging;

accessing data related to an operation of the target construct by a debug construct in real time;

monitoring at least a portion of the accessed data without disturbing the operation of the target construct; and

debugging the target construct using the monitored portion of the accessed data.

2. (Original) The method of claim 1 further comprising modifying at least a portion of the data.

3. (Original) The method of claim 1 wherein the target construct is one selected from the group consisting of a service, a socket, a service stack, a set of services, and a set of sockets.

4. (Original) The method of claim 1 wherein the debug construct comprises at least one service, at least one socket,

**Appln No. 09/618,965**

**Amdt date May 28, 2004**

**Reply to Office action of April 2, 2004**

B1  
or a combination of at least one service and at least one socket.

5. (Original) The method of claim 1 wherein selecting a target construct further comprises:

providing information about a plurality of services; and  
selecting the target construct from the plurality of services.

6. (Original) The method of claim 5 wherein the information includes a current state of each of the plurality of services.

7. (Original) The method of claim 5 further comprising:

providing information about a plurality of sockets;  
and  
selecting the target construct from the plurality of sockets.

8. (Original) The method of claim 7 wherein the information includes a current state of each of the plurality of services.

9. (Original) The method of claim 1 further comprising accessing a memory of the target construct by the debug construct, the accessing corresponding to reading the memory or writing to the memory.

10. (Original) The method of claim 1 further comprising accessing state of the target construct by the debug

**Appln No. 09/618,965**

**Amdt date May 28, 2004**

**Reply to Office action of April 2, 2004**

31  
construct, the accessing corresponding to reading the state or modifying the state.

11. (Original) The method of claim 1 further comprising dynamically allocating the debug construct.

12. (Original) The method of claim 1 further comprising dynamically de-allocating the debug construct once the monitoring is completed.

13. (Original) The method of claim 1 further comprising collecting statistics related to the target construct.

14. (Original) The method of claim 1 further comprising transmitting the data to at least one host system.

15. (Original) The method of claim 14 wherein the data is transmitted based upon a request sent by a host application.

16. (Original) The method of claim 14 wherein an operating system determines which data is to be transmitted.

17. (Original) The method of claim 14 wherein the debug construct specifies which data is to be transmitted.

18. (Original) The method of claim 1 further comprising notifying the debug construct upon a completion of a certain operation by the target construct.

19. (Original) The method of claim 14 further comprising:

**Appln No. 09/618,965**

**Amdt date May 28, 2004**

**Reply to Office action of April 2, 2004**

B1  
measuring bandwidth required to transmit the data; and  
transmitting at least a portion of data based upon  
available bandwidth.

20. (Original) The method of claim 1 wherein debugging  
is performed in a multi-channel, multi-service environment.

21. (Original) The method of claim 15 wherein sending  
the request and transmitting the response are performed over a  
network.

22. (Original) The method of claim 1 further  
comprising:

collecting at least a portion of the data;  
allocating a copy of the target construct in a  
simulated environment; and  
debugging the operation of the target construct using  
the collected data in the simulated environment.

23. (Original) The method of claim 1 further  
comprising:

generating a request by a host application;  
transmitting the request to an operating system;  
performing the request by the operating system; and  
sending a response to the host application.

24. (Currently Amended) A method for multi-channel, multi-  
service debugging, comprising:

providing information about a plurality of running  
services;

maintaining an isolated debugging environment for each  
of the plurality of running services, wherein the isolated

Appln No. 09/618,965

Amdt date May 28, 2004

Reply to Office action of April 2, 2004

B1  
debugging environment provides a separate context for each running service; and

selecting a target construct for debugging from the plurality of running services; and

dynamically loading one or more of the plurality of running services into the target construct.

25. (Previously Presented) The method of claim 24 wherein the information about the plurality of running services includes a current state of each service.

26. (Original) The method of claim 24 further comprising:

providing information about at least one socket;  
maintaining an isolated debugging environment for each of the at least one socket; and

selecting a target construct for debugging from the at least one socket.

27. (Original) The method of claim 26 wherein the information about the at least one socket includes a current state of each socket.

28. (Original) The method of claim 24 wherein the target construct is one selected from the group consisting of a service, a socket, a service stack, a set of services, and a set of sockets.

29. (Original) The method of claim 28 further comprising switching between services and sockets during a debugging process.

**Appln No. 09/618,965**

**Amdt date May 28, 2004**

**Reply to Office action of April 2, 2004**

B1  
30. (Original) The method of claim 24 wherein the isolated debugging environment is maintained by an operating system in cooperation with a host application.

31. (Original) The method of claim 24 wherein the target construct is selected based upon a request from a host application.

32. (Original) The method of claim 24 further comprising:

- generating a request by a host application;
- transmitting the request to an operating system;
- performing the request by the operating system; and
- sending a response to the host application.

33. (Original) The method of claim 32 wherein transmitting the request and sending a response are performed over a network.

34. (Original) The method of claim 24 further comprising:

- sending a request by a host application; and
- receiving a response by the host application once a requested operation is completed.

35. (Original) The method of claim 34 wherein sending a request and receiving a response are performed over a network.

36. (Original) The method of claim 24 further comprising:

- receiving a request by an operating system;
- performing a requested operation; and

Appln No. 09/618,965

Amdt date May 28, 2004

Reply to Office action of April 2, 2004

transmitting a response once the requested operation is completed.

37. (Original) The method of claim 36 wherein receiving a request and transmitting a response are performed over a network.

38. (Original) The method of claim 24 further comprising dynamically allocating at least one service into the target construct.

39. (Original) The method of claim 38 further comprising instantiating any of at least one service, at least one service stack, and at least one socket.

40. (Original) The method of claim 24 further comprising substituting input and output data for at least one socket.

41. (Original) The method of claim 40 further comprising:

collecting data for at least one socket;  
allocating a copy of the target construct in a simulated environment; and  
debugging the operation of the target construct using the collected data.

42. (Currently Amended) An apparatus for interactive debugging comprising:

means for dynamically allocating a plurality of services to a plurality of processors in the multi-channel, multi-service system for execution;

Appln No. 09/618,965

Amdt date May 28, 2004

Reply to Office action of April 2, 2004

B1  
means for selecting a target construct from the dynamically allocated executing plurality of services for debugging;

means for accessing data related to an operation of the target construct by a debug construct in real time;

means for monitoring at least a portion of the accessed data without disturbing the operation of the target construct; and

debugging the target construct using the monitored portion of the accessed data.

43. (Currently Amended) An apparatus for multi-channel, multi-service debugging, comprising:

means for providing information about a plurality of running services;

means for maintaining an isolated debugging environment for each of the plurality services, wherein the isolated debugging environment provides a separate context for each running service; and

means for selecting a target construct for debugging from the plurality of running services; and

means for dynamically loading one or more of the plurality of running services into the target construct.

44. (Currently Amended) An apparatus for interactive debugging in a multi-channel, multi-service system comprising:

a plurality of services dynamically allocated to a plurality of processors in the multi-channel, multi-service system for execution;

a target construct selected from the dynamically allocated executing plurality of services; and



**Appln No. 09/618,965**

**Amdt date May 28, 2004**

**Reply to Office action of April 2, 2004**

31  
a debug construct configured to access data related to an operation of the target construct in real time and to monitor at least a portion of the data without disturbing the operation of the target construct.

45. (Original) The apparatus of claim 44 wherein the debug construct is further configured to modify at least a portion of the data.

46. (Original) The apparatus of claim 44 wherein the target construct is one selected from the group consisting of a service, a socket, a service stack, a set of services, and a set of sockets.

47. (Original) The apparatus of claim 44 wherein the debug construct comprises at least one service, at least one socket, or a combination of at least one service and at least one socket.

48. (Original) The apparatus of claim 44 further comprising a user interface for providing information about a plurality of services and selecting the target construct from the plurality of services upon a user request.

49. (Original) The apparatus of claim of claim 48 wherein the information about a plurality of services includes a current state of each of the plurality of services.

50. (Original) The apparatus of claim 48 wherein the user interface further provides information about a plurality of sockets and allows the user to select the target construct from the plurality of sockets.

**Appln No. 09/618,965**

**Amdt date May 28, 2004**

**Reply to Office action of April 2, 2004**

B1  
51. (Original) The apparatus of claim of claim 50 wherein the information about a plurality of sockets includes a current state of each of the plurality of sockets.

52. (Original) The apparatus of claim 48 wherein the user interface is a text-based interface or graphical user interface.

53. (Original) The apparatus of claim 44 further comprising a platform control socket configured to dynamically allocate the debug construct.

54. (Original) The apparatus of claim 44 further comprising a platform control socket further configured to dynamically de-allocate the debug construct once the monitoring is completed.

55. (Original) The apparatus of claim 44 further comprising a profiler collecting statistics related to the target construct.

56. (Original) The apparatus of claim 44 further comprising:

at least one host processor; and  
a communications infrastructure for transmitting the data to the host processor.

57. (Original) The apparatus of claim 56 further comprising an operating system configured to determine which data is to be transmitted, measure bandwidth required to transmit the data, and determine a portion of the data to be transmitted based upon available bandwidth.

**Appln No. 09/618,965**

**Amdt date May 28, 2004**

**Reply to Office action of April 2, 2004**

B1  
58. (Original) The apparatus of claim 56 wherein the debug construct is further configured to specify which portion of the data is to be transmitted.

59. (Original) The apparatus of claim 56 wherein the data is transmitted based upon the request sent by a host application.

60. (Original) The apparatus of claim 44 wherein debugging is performed in a multi-channel, multi-service environment.

61. (Original) The apparatus of claim 56 further comprising:

- a host application generating a request;
- a communications infrastructure transmitting the request to the debug construct; and
- the debug construct configured to perform the request and to send a response to the host application.

62. (Original) The apparatus of claim 61 wherein the communications infrastructure is a network.

63. (Original) The apparatus of claim 56 further comprising a host application sending a request and receiving a response once a requested operation is completed.

64. (Original) The apparatus of claim 63 wherein the host application sends a request and receives a response over a network.

Appln No. 09/618,965

Amdt date May 28, 2004

Reply to Office action of April 2, 2004

B1  
65. (Original) The apparatus of claim 56 wherein the debug construct is further configured to receive a request, perform a requested operation, and transmit a response once the requested operation is completed.

66. (Original) The apparatus of claim 65 wherein the debug construct receives the request and transmits the response over a network.

67. (Currently Amended) An apparatus for multi-channel, multi-service debugging, comprising:

a graphical user interface for providing information about a plurality of running services;

an operating system maintaining an isolated debugging environment for each of the plurality of running services, wherein the isolated debugging environment provides a separate context for each running service; and

a debug core configured to select a target construct for debugging from the plurality of running services upon a user request; and

means for dynamically loading one or more of the plurality of running services into the target construct.

68. (Original) The apparatus of claim 67 wherein the information about the at least one service includes a current state of each service.

69. (Original) The apparatus of claim 67 wherein the graphical user interface provides information about at least one socket, the operating system maintains an isolated debugging environment for each of the at least socket, and the debug core

**Appln No. 09/618,965**

**Amdt date May 28, 2004**

**Reply to Office action of April 2, 2004**

is configured to select a target construct for debugging from the at least one socket upon a user request.

70. (Original) The apparatus of claim 69 wherein the information about the at least one socket includes a current state of each socket.

71. (Original) The apparatus of claim 67 wherein the target construct is one selected from the group consisting of a service, a socket, a service stack, a set of services, and a set of sockets.

72. (Original) The apparatus of claim 67 wherein the debug core is further configured to switch between services and sockets during a debugging process upon a user request.

73. (Original) The apparatus of claim 67 further comprising a host application configured to send a request to select the target construct.

74. (Original) The apparatus of claim 73 further comprising:

a communications infrastructure transmitting the request to an operating system; and

the operating system configured to perform the request.

75. (Original) The apparatus of claim 74 wherein the communications infrastructure is a network.

76. (Original) The apparatus of claim 67 further comprising a host application sending a request for a debugging

Appln No. 09/618,965

Amdt date May 28, 2004

Reply to Office action of April 2, 2004

operation and receiving a response once the operation is completed.

B1  
77. (Original) The apparatus of claim 67 wherein the operating system receives a request for a debugging operation, performs the operation, and transmits a response once the requested operation is completed.

78. (Original) The apparatus of claim 67 further comprising a host application requesting to dynamically allocate at least one service into the target construct and to instantiate at least one service or at least one service stack.

79. (Original) The apparatus of claim 67 wherein a host application cooperates with the operating system to substitute input and output data for at least one socket.

80. (Original) The apparatus of claim 79 wherein the host application is configured to request to collect data for at least one socket, to allocate a copy of the target construct in a simulated environment, and to debug the operation of the target construct using the collected data in the simulated environment.

81. (Cancelled).

82. (Cancelled).

83. (Currently Amended) A computer readable medium comprising instructions, which when executed on a processor, perform a method for interactive debugging in a multi-channel, multi-service system, the method comprising:

Appln No. 09/618,965

Amdt date May 28, 2004

Reply to Office action of April 2, 2004

B1  
dynamically allocating a plurality of services to a plurality of processors in the multi-channel, multi-service system for execution;

selecting a target construct for debugging from the dynamically allocated executing plurality of services;

accessing data related to an operation of the target construct by a debug construct in real time;

monitoring at least a portion of the accessed data without disturbing the operation of the target construct; and

debugging the target construct using the monitored portion of the accessed data.

84. (Currently Amended) A computer readable medium comprising instructions, which when executed on a processor, perform a method for multi-channel, multi-service debugging, comprising:

providing information about a plurality of running services;

maintaining an isolated debugging environment for each of the plurality of running services, wherein the isolated debugging environment provides a separate context for each running service; and

selecting a target construct for debugging from the plurality of running services; and

dynamically loading one or more of the plurality of running services into the target construct.

---